UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/527,861	03/16/2005	Christian Hentschel	NL 020911	2056

24737   7590   09/24/2009
PHILIPS INTELLECTUAL PROPERTY & STANDARDS
P.O. BOX 3001
BRIARCLIFF MANOR, NY 10510

EXAMINER
TSAI, SHENG JEN

ART UNIT	PAPER NUMBER
2186	

MAIL DATE	DELIVERY MODE
09/24/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Application Number: 10/527,861
Filing Date: March 16, 2005
Appellant(s): HENTSCHEL, CHRISTIAN

Edward W. Goodman
<u>For Appellant</u>

EXAMINER'S ANSWER

This is in response to the appeal brief filed 8/10/2009 appealing from the Office action

mailed 3/17/2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Mater

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal Identifies the ground of rejections and the associated claims under rejection to be reviewed on appeal.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

US 6,091,414	Kraft, IV et al.	07-18-2000
US 7,200,857	Rodriguez et al.	04-03-2007

US 5,596,502   Koski et al.   01-21-1997

(9) Grounds of Rejection

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

>>>   Claims 1, 3-4, 6-7, 9, 11-12, 14-15 and 17 are rejected under 35 U.S.C. 103(a)

as being unpatentable over Kraft, IV et al. (US 6,091,414, hereinafter referred to as

Kraft), in view of Rodriguez et al. (US 7,200,857, hereinafter referred to as Rodriguez).

It is noted that, in the following claim analysis, those elements recited by the

claims are presented using **bold** font.

As to claim 1, Kraft discloses **a method** [System and Method for Cross-

Environment Interaction in a Computerized Graphical Interface Environment (title)] **of**

allocating shared resources [the shared resources includes the displaying screen

(figure 3 shows that two windows (30 and 35) associated two different applications are

sharing the same display screen (26) of the computer system) and CPU time (this, in

turn, will cause the system to increase CPU utilization of task 1 relative to task 2,

column 7, lines 41-42)] **between applications with media information on a resource**

limited platform [the X Window System environment permits execution of applications

in the context of various windows (column 1, lines 63-64); A system and method for

automatically adjusting priority assigned to execution of applications, tasks, or

workspaces (abstract); figure 3 shows that two windows (30 and 35) associated two

different applications are sharing the same display screen (26) of the computer system;

When a given application such as X application 24 goes into and out of "focus", this

may be visually represented by an associated window such as window 30 in display

26, by means of the border of the window changing color, and the window associated

with the particular application coming to the top or being placed lower in the stacking

order, respectively (column 5, lines 58-64)], **characterized in that the method**

comprises the following steps:

identifying an application with a current focus of a user [In response to this user

input, detected by the X Server 28, the X Server will communicate this input to the

Window Manager 22, thereby indicating that window 35 has gained focus (column 7,

lines 53-56)];

setting or increasing the allocation of resources for the application with the

current focus of the user [it will be recalled that an additional feature of the invention

is to provide for additional CPU 10 utilization for the task, application, or suite of

applications associated with that "focused" window (column 5, line 67 to column 6, line

3); The amount of CPU resource then directed to the particular application as a result

of the priority alteration is thereby in turn altered. In this manner, a focused application

is dynamically provided with more CPU resource relative to remaining tasks,

applications, or suites thereof associated with a workspace executing in the

multitasking environment (column 4, lines 17-23)]; **and automatically allocating a**

remaining part of the resources to at least one application without the current

focus of the user [Thus, the system in combination with the software steps

represented in FIG. 4 will cause the appearance of a window associated with the task 1

to be changed and brought to the foreground, thereby signifying that the task 1 is

focused. The system will then set the focus of task 1. This, in turn, will cause the

system to increase CPU utilization of task 1 relative to that of task 2. Next, the system

will alter the windowed appearance of the task 2 and move it to the background. Next,

the system will clear focus of the previously selected task 2 and its corresponding

window. CPU utilization for task 2 will then be decreased to that of task 1 prior to its

selection (column 7, lines 36-47); hence, task 1 is in focus and task 2 is out of focus,

but task2 is still allocated to certain amount of CPU time, and task 2 is also allocated to

be displayed in the background of the screen], **wherein the step of identifying the**

application with the current focus of the user is selected from at least one of the

group of: user controlled, system controlled, or externally controlled [Kraft

teaches that the current focus of the user is selected, at least, via user controller

mechanism: It will first be assumed that a user input is typically provided by means of

the keyboard 34 or mouse 36, reflecting that a particular window and its associated

task is of more interest or priority to the user. This is shown graphically in FIG. 3 by the

arrow 33, indicating that a cursor of the mouse pointing device 36 for example, has

been moved by a user into the window 35. This indicates that the end user desires to

cause the window 35 to be "focused" into prominence both visually in the display 26

and in terms of the processing power allocated to tasks associated with this window 35

by the CPU 10 (column 6, lines 55-65); First it is assumed that the user has moved the

pointing device, 51, whereby a window is either selected, 52, or deselected, 57.

Assuming a window was selected, the window manager will then set focus (column 7,

lines 12-15); In response to this user input, detected by the X Server 28, the X Server

will communicate this input to the Window Manager 22, thereby indicating that window

35 has gained focus (column 7, lines 53-56)].

As to claim 1, Kraft does not teach that **a provider of the media information**

performs the externally controlled step of identifying the application with the

current focus of the user.

However, Rodriguez teaches in the invention "Synchronized Video-On-Demand

Supplemental Commentary" a scheme for allowing multiple applications/windows to

share display screen [The window manager 59 provides a mechanism for

implementing the sharing of the screen regions and user input (column 5, lines 28-30)],

which is very similar to the invention disclosed by Kraft.

Specifically, Rodriguez teaches a provider of the media information performs the

externally controlled step of identifying the application with the current focus of the user

[Advertisement pop-ups are optional supplements that enable the user to receive

product information during the on-demand video presentation in exchange for a lower

rental fee. The advertisements may be specific to products inherent during the course

of the on-demand movie such as a watch worn by a leading actor. Furthermore, the

advertisement supplement's active time interval may be concurrent to the time in the

movie that the leading actor reads the time on the watch (column 10, lines 12-20); note

that Advertisement pop-ups are externally controlled and provided by a media provider;

figure 7 of Rodriguez shows the display screen with "pop-up comments" (123) as one of

the displayed item. Significantly, it specifically shows that the "pop-up comments" is the

high-lighted item (with board outlines), which indicates it is the current focus of the user;

figure 8 further shows the display of the screen when the "pop-up comments" is

activated -- FIG. 8 depicts an example stopped video window 130 that is presented to

the user after the user stops the presentation of a video rental for which actors' pop-up

comments had been activated ... Rental control options list 133 contains rental control

options, such as the option to "De-activate comments" 134 ... (col. 11, lines 14-34),

Note that activating and de-activating of the "pop-up comments" further confirm that it is

the current focus of the user; Further, figures 9 and 10 also show the display of the

screen when the "pop-up comments" is activated -- FIG. 9 depicts an example stopped

video window 140 that is presented to the user after the user stops the presentation of a

video rental for which pop-up comments are not activated. Rental control options list

133 contains rental control options, such as the option to "Activate pop-up comments"

144, and a highlighted option area 135. A user can activate pop-up comments by

selecting the "Activate pop-up comments" option 144 via the remote control device 80

... (col. 11, lines 35-45); FIG. 10 depicts an example pop-up comments selection

window 150 that is presented to the user after the user selects the "Activate pop-up

comments" option 144 via stopped video window 140 ... (col. 11, lines 61-67)].

Rodriguez also teaches that the motivation of allowing a provider of the media

information performs the externally controlled step of identifying the application with the

current focus of the user is to give users the option of saving [Advertisement pop-ups

are optional supplements that enable the user to receive product information during the

on-demand video presentation in exchange for <u>a lower rental fee</u> (column 10, lines 12-

15)].

Therefore, it would have been obvious for one of ordinary skills in the art at the

time of Applicants' invention to allow a provider of the media information performs the

externally controlled step of identifying the application with the current focus of the

user, as demonstrated by Rodriguez, and to incorporate it into the existing scheme

disclosed by Kraft, in order to give users more options and more satisfaction.

As to claim 3, Kraft teaches **the method as claimed in claim 1, characterized**

in that the user controlled step of identifying the application with the current

focus of the user, comprises one or more of the following steps:

selecting a new application as the application with the current focus of the user,

when the new application is opened [In an effort to address this deficiency regarding

distributive applications, in yet an alternate implementation, <u>newly built (e.g., non-</u>

<u>legacy) applications</u> may be linked to an Xt Intrinsics library capable of reprioritizing

themselves as they <u>leave or enter focus</u> (column 9, lines 24-28)];

changing the application with the current focus of the user to an application just

switched to upon switching to an already opened application [It will first be

assumed that <u>a user input</u> is typically provided by means of the <u>keyboard 34 or mouse</u>

<u>36</u>, reflecting that a particular window and its associated task is of more interest or

priority to the user. This is shown graphically in FIG. 3 by the arrow 33, indicating that

a cursor of the mouse pointing device 36 for example, has been moved by a user into

the window 35. This indicates that the end user desires to cause the window 35 to be

"focused" into prominence both visually in the display 26 and in terms of the processing

power allocated to tasks associated with this window 35 by the CPU 10 (column 6,

lines 55-65)];

when the user closes down an application with the current focus, switching to

the application with the preceding focus of the user by keeping a record of the

order of previously opened applications to indicate their importance in

decreasing order, where the most recently opened application has the highest

importance [Thus, the system in combination with the software steps represented in

FIG. 4 will cause the appearance of a window associated with the task 1 to be changed

and brought to the foreground, thereby signifying that the task 1 is focused. The

system will then set the focus of task 1. This, in turn, will cause the system to increase

CPU utilization of task 1 relative to that of task 2. Next, the system will alter the

windowed appearance of the task 2 and move it to the background. Next, the system

will clear focus of the previously selected task 2 and its corresponding window. CPU

utilization for task 2 will then be decreased to that of task 1 prior to its selection

(column 7, lines 36-47); When a given application such as X application 24 goes into

and out of "focus", this may be visually represented by an associated window such as

window 30 in display 26, by means of the border of the window changing color, and the

window associated with the particular application coming to the top or being placed

lower in the stacking order, respectively. In addition to this window manager 22 setting

or clearing focus so as to cause the GUI window of the related application to come to the top of or being placed within the display stack 26 (column 5, lines 58-66)].

As to claim 4, Kraft teaches **the method as claimed in claim 1, characterized in that the system controlled step of identifying the application with the current focus of the user, is performed by one of the following steps**: **an automatically changing of the current focus of the user according to a predetermined priority hierarchy of the available applications** [A system and method are provided for automatically adjusting priority assigned to execution of applications, tasks, or workspaces to thereby improve performance relative to other such applications, tasks or workspaces in a computerized multitasking graphical user interface environment (column 3, lines 61-66); The amount of CPU resource then directed to the particular application as a result of the priority alteration is thereby in turn altered. In this manner, a focused application is dynamically provided with more CPU resource relative to remaining tasks, applications, or suites thereof associated with a workspace executing in the multitasking environment (column 4, lines 17-23)]; **keeping a record of the order of previously opened applications to indicate their importance in decreasing order, where the most recently opened application has the highest importance and, switching to the application with the preceding focus of the user when the user closes down an application with the current focus** [the record includes display stack (figure 3, 26) and window stacks; When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by

means of the border of the window changing color, and the window associated with the

particular application <u>coming to the top or being placed lower in the stacking order,</u>

<u>respectively</u>. In addition to this window manager 22 setting or clearing focus so as to

cause the GUI window of the related application to come to the top of or being placed

within <u>the display stack</u> 26 (column 5, lines 58-66); Thus, the system in combination

with the software steps represented in FIG. 4 will cause the appearance of a window

associated with the task 1 to be changed and brought to the foreground, thereby

signifying that the <u>task 1 is focused</u>. The system will then <u>set the focus of task 1</u>. This,

in turn, will cause the system to increase CPU utilization of <u>task 1</u> relative to that of

<u>task 2</u>. Next, the system will <u>alter the windowed appearance of the task 2 and move it</u>

<u>to the background</u>. Next, the system will clear focus of the previously selected task 2

and its corresponding window. <u>CPU utilization for task 2 will then be decreased</u> to that

of task 1 prior to its selection (column 7, lines 36-47)].

As to claim 6, Kraft teaches **the method as claimed in claim 1, characterized**

in that the step of setting or increasing the output quality of the application with

current focus of the user is performed automatically by means of automatic

settings of the overall system control and with no additional input from the user

[A system and method are provided for <u>automatically adjusting priority</u> assigned to

execution of applications, tasks, or workspaces to thereby improve performance

relative to other such applications, tasks or workspaces in a computerized multitasking

graphical user interface environment (column 3, lines 61-66); The amount of CPU

resource then directed to the particular application as a result of the priority alteration is

thereby in turn altered. In this manner, <u>a focused application</u> is <u>dynamically provided with more CPU resource relative to remaining tasks</u>, applications, or suites thereof associated with a workspace executing in the multitasking environment (column 4, lines 17-23); When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by means of <u>the border of the window changing color</u>, and the window associated with the particular application coming to the top or being placed lower in the stacking order, respectively (column 5, lines 58-64)].

As to claim 7, Kraft teaches **the method as claimed in claim 1** [refer to "As to claim 1"], **characterized in the step of decreasing or increasing the allocation of resources for the application with the current focus of the user is performed manually by user interaction by means of a user interface** [It will first be assumed that <u>a user input</u> is typically provided by means of the <u>keyboard 34 or mouse 36</u>, reflecting that a particular window and its associated task is of more interest or priority to the user. This is shown graphically in FIG. 3 by the arrow 33, indicating that a cursor of the mouse pointing device 36 for example, has been moved by a user into the window 35. This indicates that the end user desires to cause the window 35 to be "focused" into prominence both visually in the display 26 and in terms of the processing power allocated to tasks associated with this window 35 by the CPU 10 (column 6, lines 55-65)].

As to claim 9, it recites substantially the same limitations as in claim 1, and is rejected for the same reasons set forth in the analysis of claim 1. Refer to "As to claim 1" presented earlier in this Office Action for details.

As to claim 10, it recites substantially the same limitations as in claim 2, and is rejected for the same reasons set forth in the analysis of claim 2. Refer to "As to claim 2" presented earlier in this Office Action for details.

As to claim 11, it recites substantially the same limitations as in claim 3, and is rejected for the same reasons set forth in the analysis of claim 3. Refer to "As to claim 3" presented earlier in this Office Action for details.

As to claim 12, it recites substantially the same limitations as in claim 4, and is rejected for the same reasons set forth in the analysis of claim 4. Refer to "As to claim 4" presented earlier in this Office Action for details.

As to claim 13, it recites substantially the same limitations as in claim 5, and is rejected for the same reasons set forth in the analysis of claim 5. Refer to "As to claim 5" presented earlier in this Office Action for details.

As to claim 14, it recites substantially the same limitations as in claim 6, and is rejected for the same reasons set forth in the analysis of claim 6. Refer to "As to claim 6" presented earlier in this Office Action for details.

As to claim 15, it recites substantially the same limitations as in claim 7, and is rejected for the same reasons set forth in the analysis of claim 7. Refer to "As to claim 7" presented earlier in this Office Action for details.

As to claim 17, Kraft teaches **a computer-readable medium having stored thereon instructions for causing a processing unit to execute the method as claimed in claim 1** [This invention relates to <u>computer systems</u> supporting graphical user interface environments and, more particularly, relates to systems and methods for obtaining information and effecting application control across dissimilar computer system boundaries (column 1, lines 14-18); FIG. 4 is <u>a flow diagram implementable in software executing on the system</u> of FIG. 3 illustrating the steps of the method of the invention (column 4, lines 37-39)].

>>> Claims 8 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kraft, IV et al. (US 6,091,414, hereinafter referred to as Kraft), in view of Rodriguez et al. (US 7,200,857, hereinafter referred to as Rodriguez), and further in view of Koski et al. (US 5,596,502, hereinafter referred to as Koski).

As to claim 8, Kraft in view of Rodriguez teaches **the method as claimed in claim 6** [refer to "As to claim 6"], but does not teach that **the automatic settings of the overall system control is influenced by a learning function, which takes previous user settings of the past into account, wherein the learning function is implemented as at least one of an averaging function, a recursive function, a non-recursive function, a non-linear function, a function with different weights, having the previous user settings as an input**.

However, Koski teaches in the invention "Computer System Including means for Decision Support Scheduling" a scheme for allocating the best available resources to produce products [in response to demand placed on the Cube World by a customer

order, which program <u>allocates the best available resources to produce products</u>

(abstract)], where a learning function is used [This invention relates generally to

computerized management systems, and more particularly to computer-controlled

closed loop multiple activity management coordinating systems that support effective

resource utilization decision making and alternate path non-deterministic process

development, including <u>rule based learning functions to efficiently allocate resources</u> to

resolve conflicts in competition for the availability of limited resources to achieve

multiple objectives scheduled in time (col. 1, lines 19-28); <u>Learning-based</u> : ... (col. 9,

line 60 to col. 10, line 15)], and the learning function comprises a function with different

weights [However, as the Cube System operates within the plant and receives prime

data, it learns that skilled operators tend to assign certain types of products to certain

machines and <u>it adjusts the weighted rankings for those paths and learns to load</u>

<u>specific machines with those types of jobs</u> (col. 9, line 65 to col. 10, line 10); <u>It</u>

<u>examines all possible alternate paths and ranks them based on desirability weights</u>.

These desirability weights may be arbitrarily assigned by the user but usually will be

learned by the system as history patterns, retained as Prime Data, are recognized by a

neural network built into Cube Objects (col. 20, lines 3-9)].

Koski also teaches that the motivation of having a learning function with different

weights to take previous user setting into account when performing automatic setting is

to allocate the best available resources [in response to demand placed on the Cube

World by a customer order, which program <u>allocates the best available resources to</u>

<u>produce products</u> (abstract)].

Therefore, it would have been obvious for one of ordinary skills in the art at the time of Applicants' invention to have a learning function with different weights to take previous user setting into account when performing automatic setting, as demonstrated by Koski, and to incorporate it into the existing scheme disclosed by Kraft in view of Rodriguez, in order to provide customized display according to user's preference.

As to claim 16, it recites substantially the same limitations as in claim 8, and is rejected for the same reasons set forth in the analysis of claim 8. Refer to "As to claim 8" presented earlier in this Office Action for details.

(10) Response to Arguments

Appellants' arguments have been fully and carefully considered with Examiner's answers set forth below.

Answer to Arguments on Claims 1 and 9

(1) Appellants contend that, with respect to claims 1 and 9, the Rodriguez reference fails to teach that there is a reallocation of the resources of the system such that the pop-up has the current focus of the user (see page 15, first paragraph of Appellants' Appeal Brief). The Examiner disagrees.

First, Appellants readily acknowledge that "there must be a reallocation of resources such that the pop-ups of Rodriguez are visible" (see page 15, last paragraph of Appellants' Appeal Brief). Thus the only remaining question regarding the cited limitation is "whether the pop-up has the current focus of the user."

Second, figure 7 of Rodriguez shows a <u>current</u> display of the Video-On-Demand screen with "pop-up comments" (123) as one of the <u>current</u> displayed items.

Significantly, it specifically shows that the "pop-up comments" is the high-lighted item

(with **board outlines**), which indicates it is the current focus of the user.

Further, figures 8-10 shows the subsequent displays of the screen when the

"pop-up comments" is activated [FIG. 8 depicts an example stopped video window 130

that is presented to the user after the user stops the presentation of a video rental for

which actors' pop-up comments had been activated ... Rental control options list 133

contains rental control options, such as the option to "De-activate comments" 134 ...

(col. 11, lines 14-34); FIG. 9 depicts an example stopped video window 140 that is

presented to the user after the user stops the presentation of a video rental for which

pop-up comments are not activated. Rental control options list 133 contains rental

control options, such as the option to "Activate pop-up comments" 144, and a

highlighted option area 135. A user can activate pop-up comments by selecting the

"Activate pop-up comments" option 144 via the remote control device 80 ... (col. 11,

lines 35-45); FIG. 10 depicts an example pop-up comments selection window 150 that

is presented to the user after the user selects the "Activate pop-up comments" option

144 via stopped video window 140 ... (col. 11, lines 61-67)].

Note that "pop-up comments" (figure 7, 123) is one of the current displayed

items, and that activating and de-activating of the "pop-up comments" requires a user to

"position" a cursor at the item of interest, then to "click" and "select" the item, a practice

that is typical and standard for interactive consumer electronic products. Indeed, the

actions of "positioning," "clicking" and "selecting" all require a user to focus attention on

the desired item so that only the correct and desired item is selected, hence confirming

that the "pop-up comments" item is indeed the current focus of the user.

Third, Appellants also argue that the fact that a user needs to click on the pop-up

item would be indicative that the pop-up item is not the current focus of the user.

However, the limitation actually recited in the claims is "identifying an application

with a current focus of a user." Refer again to figure 7 of Rodriguez, which shows a

display of a current Video-On-Demand screen with two menus. Even if, for the sake of

discussion without conceding the point, one assumes that the "current" focus of the user

is limited to only the "current" display of the screen and not the "subsequent" displays as

shown in figures 8-10 following the clicking and selecting, it is still a fact that the "pop-

up" item is, among others, part of the "current" display as shown in figure 7. Further, the

clicking and selecting of the "pop-up" item represents the action of "identifying an

application with a current focus of the used," in which the application identified is the

corresponding "pop-up" item.

Therefore, Rodriguez teaches the cited limitation of "a reallocation of the

resources of the system such that the pop-up has the current focus of the user" and

"identifying an application with a current focus of the used."

Therefore, the Examiner maintains the rejections of claims 1 and 9 as being

unpatentable.

Answer to Arguments on Claims 8 and 16

(1) Appellants contend, with respect to claims 8 and 16, that the Koski reference

fails to mitigate the deficiency of the Kraft and the Rodriguez references regarding the

limitations "a reallocation of the resources of the system such that the pop-up has the current focus of the user" and "identifying an application with a current focus of the used" as recited in claims 1 and 9. Thus claims 8 and 16, which depend from claims 1 and 9, respectively, should be allowable.

However, the Examiner has demonstrated and explained in the preceding section that the Rodriguez reference indeed teaches those limitations recited in claims 1 and 9. Refer to "**Answer to Arguments on Claims 1 and 9**" presented earlier in this Examiner's Answer for details.

Therefore, the Examiner maintains the rejections of claims 8 and 16 as being unpatentable.

(11) Related Proceedings Appendix

There are no decisions rendered by a court or the Board that may directly affect, be affected by, or have a bearing on the decision of the Board in the instant appeal.

/Sheng-Jen Tsai/ Primary Examiner, Art Unit 2186
/Matt Kim/ Supervisory Patent Examiner, Art Unit 2186
/Kevin L Ellis/ Supervisory Patent Examiner, Art Unit 2117

September 21, 2009